CLAIMS

What is claimed is:

1. A method for performing a distributed transaction in a shared-nothing database system, the method comprising:

on a first shared-nothing node of said shared-nothing database system, causing a coordinator that is coordinating the distributed transaction to store information that indicates status of said distributed transaction on a persistent storage device;

wherein the persistent storage device is accessible to a participant that is to perform one or more operations as part of said distributed transaction;

wherein the participant resides on a second shared-nothing node of said shared-nothing database system; and

on the second shared-nothing node of said shared-nothing database system, causing the participant to determine the status of said distributed transaction by reading the information from the persistent storage device.

2. The method of Claim 1 wherein:

the participant is a first participant of a plurality of participants in said distributed transaction;

the plurality of participants includes a second participant that does not have access to said persistent storage device; and

the method further comprises the step of the coordinator interacting with the second participant according to a two-phase commit protocol.

3. The method of Claim 1 further comprising the steps of:

the coordinator committing the distributed transaction;

after the coordinator commits the distributed transaction, the coordinator sending a commit message to the participant; and

preventing the information that indicates the status of the distributed transaction from being overwritten or deleted until a set of conditions is satisfied, wherein one condition in said set of conditions is that the coordinator receives a commit acknowledge message from said participant.

4. The method of Claim 1 further comprising the steps of:

the participant sending a first piece of information to the coordinator, wherein the first piece of information is associated with work performed by said participant as part of said distributed transaction; and

the coordinator performing a comparison between the first piece of information and information associated with a redo log of said second shared-nothing node; and

the coordinator determining whether to commit the transaction based, at least in part, on said comparison.

5. The method of Claim 4 wherein piece of information includes a log-sequence-number of the latest change made by the participant as part of the distributed transaction.

6. The method of Claim 5 wherein the step of sending includes the steps of:

the participant identifying a message that is being sent to said first shared-nothing node for a purpose unrelated to the distributed transaction; and

piggybacking the log-sequence number on said message.

7. A method for performing a distributed transaction in a shared-nothing database system, the method comprising:

assigning a participant to perform one or more operations as part of said distributed transaction;

wherein the participant resides on a first shared-nothing node of said shared-nothing system;

causing said participant to store, on a persistent storage device, status information that indicates changes made by the participant during performance of said one or more operations;

wherein the persistent storage device is accessible to a coordinator that is responsible for coordinating said distributed transaction;

wherein the coordinator resides on a second shared-nothing node of said shared-nothing database system;

on said second shared-nothing node of said shared-nothing database system, causing said coordinator to determine, based on the status information on said persistent storage device, whether the participant has written to persistent storage changes produced by performance of the one or more operations; and

the coordinator process determining whether the distributed transaction can be committed based, at least in part, on whether the participant has written to persistent storage changes produced by performance of the one or more operations.

8. The method of Claim 7 wherein:

the step of causing said participant to store, on a persistent storage device, status information that indicates changes made by the participant during performance of said one or more operations includes

causing said participant to store redo information in a redo log on said persistent storage device; and

the step of causing said coordinator to determine, based on the status information on said persistent storage device, whether the participant has written to persistent storage changes produced by performance of the one or more operations includes

inspecting the redo log of the participant to determine whether the redo information for said changes have been written to said persistent storage.

9. The method of Claim 7 wherein:

the participant is a first participant of a plurality of participants in said distributed transaction;

the plurality of participants includes a second participant that stores status information on a second persistent storage device that is not accessible by said coordinator; and

the method further comprises the step of the coordinator interacting with the second participant according to a two-phase commit protocol.

10. The method of Claim 7 wherein:

the information on said persistent storage device indicates that the participant has not written to persistent storage changes produced by performance of the one or more operations; and

the method further comprises the coordinator sending a force redo message to the participant to cause the participant to write to persistent storage the changes produced by performance of the one or more operations.

11. The method of Claim 10 wherein the step of sending a force redo message includes the steps of:

identifying a message that is being sent to said first shared-nothing node for a purpose unrelated to the distributed transaction; and

piggybacking the force redo message on said message.

12. A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 1.

13. A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 2.

14. A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 3.

15. A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 4.

16. A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 5.

17. A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 6.

18. A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 7.

19. A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 8.

20. A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 9.

21. A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 10.

22. A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 11.